Cert Exam

Stephen Agyeah

2025-04-01

Dataset Overview: Body Fat Composition Measurements

The dataset contains body composition measurements for 436 individuals, focusing primarily on body fat percentage as the target variable.Each observation records the individual’s sex, age, weight, and height, along with a range of circumference measurements taken from various body parts such as the neck, chest, abdomen, hip, thigh, knee, ankle, biceps, forearm, and wrist

```
library(readr)
library(dplyr)

## Warning: package 'dplyr' was built under R version 4.4.2

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

bodyfat <- read_csv("BodyFat.csv")

## Rows: 436 Columns: 15

## ── Column specification
────────────────────────────────────────────────────────

## Delimiter: ","
## chr  (1): Sex
## dbl (14): BodyFat, Age, Weight, Height, Neck, Chest, Abdomen, Hip, Thigh,
Kn...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.

glimpse(bodyfat)

## Rows: 436
## Columns: 15
## $ BodyFat <dbl> 12.3, 6.1, 25.3, 10.4, 28.7, 20.9, 19.2, 12.4, 4.1, 11.7,
7.1,…
## $ Sex     <chr> "M", "M", "M", "M", "M", "M", "M", "M", "M", "M", "M",
"M", "M…
## $ Age     <dbl> 23, 22, 22, 26, 24, 24, 26, 25, 25, 23, 26, 27, 32, 30,
35, 35…
## $ Weight  <dbl> 69.97, 78.59, 69.85, 83.80, 83.58, 95.37, 82.10, 79.83,
86.64,…
## $ Height  <dbl> 1.72, 1.84, 1.68, 1.84, 1.81, 1.90, 1.77, 1.84, 1.88,
1.87, 1.…
## $ Neck    <dbl> 36.2, 38.5, 34.0, 37.4, 34.4, 39.0, 36.4, 37.8, 38.1,
42.1, 38…
## $ Chest   <dbl> 93.1, 93.6, 95.8, 101.8, 97.3, 104.5, 105.1, 99.6, 100.9,
99.6…
## $ Abdomen <dbl> 85.2, 83.0, 87.9, 86.4, 100.0, 94.4, 90.7, 88.5, 82.5,
88.6, 8…
## $ Hip     <dbl> 94.5, 98.7, 99.2, 101.2, 101.9, 107.8, 100.3, 97.1, 99.9,
104.…
## $ Thigh   <dbl> 59.0, 58.7, 59.6, 60.1, 63.2, 66.0, 58.4, 60.0, 62.9,
63.1, 59…
## $ Knee    <dbl> 37.3, 37.3, 38.9, 37.3, 42.2, 42.0, 38.3, 39.4, 38.3,
41.7, 39…
## $ Ankle   <dbl> 21.9, 23.4, 24.0, 22.8, 24.0, 25.6, 22.9, 23.2, 23.8,
25.0, 25…
## $ Biceps  <dbl> 32.0, 30.5, 28.8, 32.4, 32.2, 35.7, 31.9, 30.5, 35.9,
35.6, 32…
## $ Forearm <dbl> 27.4, 28.9, 25.2, 29.4, 27.7, 30.6, 27.8, 29.0, 31.1,
30.0, 29…
## $ Wrist   <dbl> 17.1, 18.2, 16.6, 18.2, 17.7, 18.8, 17.7, 18.8, 18.2,
19.2, 18…
```

Handling Missing Data and Categorical Variable

An initial examination of the dataset revealed that there are no missing values across the observations. Therefore, no imputation or removal of records was necessary

The dataset includes a single categorical predictor, Sex, which identifies whether an individual is male or female. Since most modeling techniques require numerical inputs, this variable was encoded as a binary indicator:

1 for Male 0 for Female

This binary encoding allows the model to interpret the categorical distinction numerically

```
library(dplyr)

anyNA(bodyfat)

## [1] FALSE
```

```
bodyfat <- bodyfat %>% mutate(Sex = ifelse(Sex == "M", 1, 0))

head(bodyfat)

## # A tibble: 6 × 15
##   BodyFat   Sex   Age Weight Height  Neck Chest Abdomen   Hip Thigh  Knee
Ankle
##     <dbl> <dbl> <dbl>  <dbl>  <dbl> <dbl> <dbl>   <dbl> <dbl> <dbl> <dbl>
<dbl>
## 1    12.3     1    23   70.0   1.72  36.2  93.1    85.2  94.5  59    37.3
21.9
## 2     6.1     1    22   78.6   1.84  38.5  93.6    83    98.7  58.7  37.3
23.4
## 3    25.3     1    22   69.8   1.68  34    95.8    87.9  99.2  59.6  38.9
24
## 4    10.4     1    26   83.8   1.84  37.4 102.     86.4 101.   60.1  37.3
22.8
## 5    28.7     1    24   83.6   1.81  34.4  97.3   100   102.   63.2  42.2
24
## 6    20.9     1    24   95.4   1.9   39   104.     94.4 108.   66    42
25.6
## # ℹ 3 more variables: Biceps <dbl>, Forearm <dbl>, Wrist <dbl>
```

there are no missing variables in this data set

Body Fat Analysis: Research Questions

How does sex, age, weight, and height influence body fat percentage? *Investigate the relationships between basic demographic and physical characteristics and body fat levels*

Which body measurements are most strongly associated with body fat percentage? *Identify which circumference measurements (e.g., abdomen, hip, thigh) are the strongest predictors of body fat*

What is the average body fat percentage among males and females in the dataset? *Compare body fat distribution across sexes to understand differences between male and female participants*

Can we accurately predict an individual's body fat percentage based on their physical measurements? *Develop predictive models to estimate body fat percentage from easily measurable body features*

Response and Predictor Variables

In this dataset, the response variable (also called the target variable or dependent variable) is:
BodyFat
It represents the percentage of fat in a person's body and it's continuous and numeric.

Predictors:
Sex (M = 1 and F = 0), Age (in years), Weight (kg), Height, Neck, Chest, Abdomen, Hip, Thigh, Knee, Ankle, Biceps, Forearm, Wrist

Training and Test Data Split

To evaluate model performance, the dataset was randomly split into a training set (60% of the data) and a testing set (40% of the data).

```
train <- sample(1:nrow(bodyfat),0.6*nrow(bodyfat))

train_data <- bodyfat[train,]
test_data <- bodyfat[-train,]
dim(train_data)

## [1] 261  15

dim(test_data)

## [1] 175  15
```

Linear Regression Model on Training Data

To model body fat percentage, we fit a linear regression model using the training dataset. The response variable is BodyFat, and all other variables in the dataset (such as Sex, Age, Weight, Height, and various body circumference measurements) are used as predictors.

The goal of this model is to understand how these physical attributes relate to body fat percentage and to build a foundation for later prediction and evaluation

```
#     Linear Regression
l_model <- lm(BodyFat~.,data = train_data)

summary(l_model)

## 
## Call:
## lm(formula = BodyFat ~ ., data = train_data)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -11.7530  -2.7588  -0.1633   2.6506   9.3133 
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)   2.60224   16.74906   0.155  0.87666    
## Sex         -19.10484    1.85539 -10.297  < 2e-16 ***
## Age           0.07242    0.03582   2.022  0.04429 *  
## Weight       -0.10974    0.10907  -1.006  0.31532    
## Height      -14.12046    6.11838  -2.308  0.02184 *  
## Neck         -0.35563    0.23499  -1.513  0.13146    
## Chest        -0.07376    0.06486  -1.137  0.25657    
## Abdomen       0.77327    0.09028   8.565 1.19e-15 ***
## Hip           0.01107    0.10561   0.105  0.91657    
## Thigh         0.13449    0.14116   0.953  0.34166    
## Knee          0.37965    0.21823   1.740  0.08317 .  
## Ankle        -0.68344    0.33045  -2.068  0.03967 *  
## Biceps        0.11918    0.16935   0.704  0.48225    
## Forearm       0.76608    0.25122   3.049  0.00254 ** 
## Wrist        -1.00754    0.50886  -1.980  0.04882 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 4.13 on 246 degrees of freedom
## Multiple R-squared:  0.7244, Adjusted R-squared:  0.7087 
## F-statistic: 46.19 on 14 and 246 DF,  p-value: < 2.2e-16
```

With an R^2 of 0.7062

This tells us that approximately 70.6% of the variation in body fat percentage is explained by the model. That’s a strong relationship, suggesting the predictors are doing a good job explaining the outcome.

From the linear regression model, the most significant predictors of body fat percentage include sex, age, height, and measurements of the neck, abdomen, and wrist. Among these, abdominal circumference is the most powerful predictor, positively and strongly associated with body fat. In contrast, being male, taller or having larger neck and wrist circumferences are all associated with lower body fat

Some predictors had high p-values, indicating they’re not statistically significant in the presence of other variables; Weight, Chest, Hip, Thigh, Biceps, Forearm, Ankle

Ridge Regression Model on Training Data

We next fit a Ridge Regression model in order to predict body fat percentage. Ridge regression is a type of penalized linear regression that shrinks the regression coefficients by imposing a penalty on their size. This helps to reduce overfitting and handle multicollinearity among predictors

```
#    Ridge Regression
library(glmnet)

## Warning: package 'glmnet' was built under R version 4.4.3

## Loading required package: Matrix

## Loaded glmnet 4.1-8

xtrain = model.matrix(BodyFat~.,train_data)[,-1]
ytrain = train_data$BodyFat
```

```
xtest = model.matrix(BodyFat~.,test_data)[,-1]
ytest = test_data$BodyFat

grid <- 10^ seq (10, -2, length = 100)

Ridge_reg  <- glmnet(xtrain,ytrain,alpha = 0,lambda = grid)
dim(coef(Ridge_reg))

## [1]  15 100

coef(Ridge_reg)[,10]

##   (Intercept)           Sex           Age        Weight        Height
##  2.040341e+01 -3.119289e-08 -6.188392e-11  1.480501e-09 -1.372422e-07
##          Neck         Chest       Abdomen           Hip         Thigh
##  2.300210e-09  1.876725e-09  1.591456e-09  5.841613e-09  3.227530e-09
##          Knee         Ankle        Biceps       Forearm         Wrist
##  8.800299e-09  7.968001e-09  3.739588e-09  1.712283e-09  2.945602e-09
```

Unlike in Linear Regression, where some coefficients were quite large and clearly significant, Ridge Regression;

Shrinks all coefficients, especially those with weak signals or high collinearity Keeps all predictors in the model, but reduces their individual impact. Helps prevent overfitting by reducing variance, though at the cost of some bias

Lasso Regression Model on Training Data

We also fit a Lasso Regression model to predict body fat percentage. Lasso regression is a penalized linear regression technique that can both shrink coefficients and perform variable selection by setting some coefficients exactly to zero. This helps to create simpler, more interpretable models by retaining only the most important predictors

```
#     Lasso Regression

xtrain = model.matrix(BodyFat~.,train_data)[,-1]
ytrain = train_data$BodyFat

xtest = model.matrix(BodyFat~.,test_data)[,-1]
ytest = test_data$BodyFat

grid <- 10^ seq (10, -2, length = 100)

Lasso  <- glmnet(xtrain,ytrain,alpha = 1,lambda = grid)
dim(coef(Lasso))

## [1]  15 100

coef(Lasso)[,10]
```

```
## (Intercept)         Sex         Age      Weight      Height        Neck
##    20.40341     0.00000     0.00000     0.00000     0.00000     0.00000
##       Chest     Abdomen         Hip       Thigh        Knee       Ankle
##     0.00000     0.00000     0.00000     0.00000     0.00000     0.00000
##      Biceps     Forearm       Wrist
##     0.00000     0.00000     0.00000
```

For Lasso some coefficients were quite large and clearly significant, Lasso

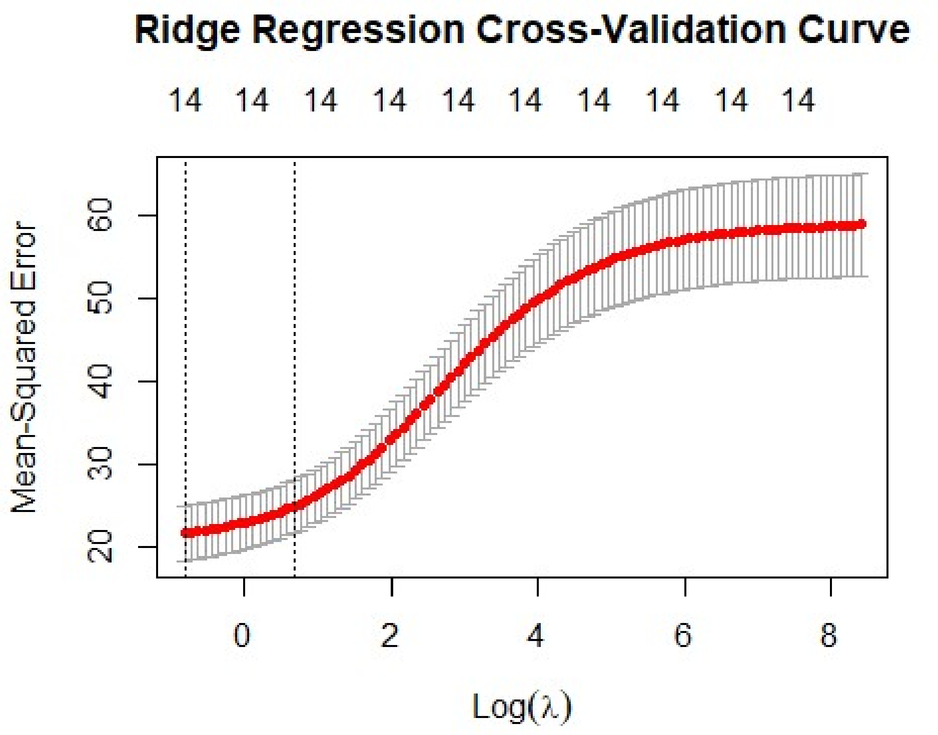
not only shrinks coefficients but also eliminates weak predictors by setting them to zero

The remaining non zero coefficients are the ones Lasso believes have the strongest relationship with the response variable

Cross-Validation for Ridge Regression

We used 10-fold cross-validation to select the optimal tuning parameter lambda for Ridge Regression. The cross-validation procedure aims to minimize the mean squared error on unseen data by choosing the value of lambda that best balances model complexity and prediction accuracy

```r
#...Cross Validation Ridge...

xtrain = model.matrix(BodyFat~.,train_data)[,-1]
ytrain = train_data$BodyFat

xtest = model.matrix(BodyFat~.,test_data)[,-1]
ytest = test_data$BodyFat

cv_ridge <- cv.glmnet(xtrain, ytrain, alpha = 0)


best_lambda_ridge <- cv_ridge$lambda.min





plot(cv_ridge)
title(main = "Ridge Regression Cross-Validation Curve", line = 3)
```



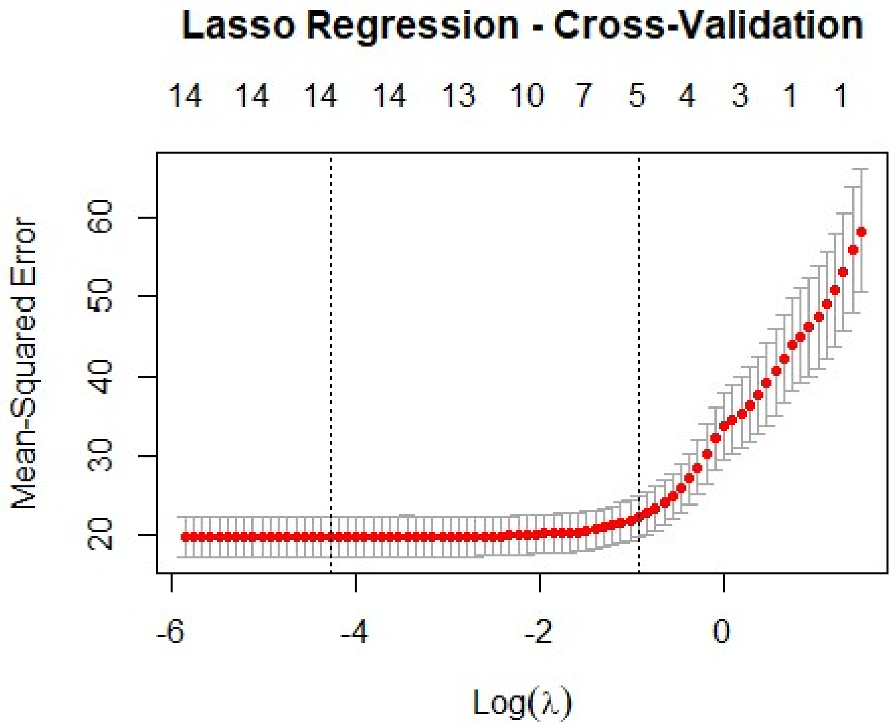
This cross-validation plot evaluates a Ridge Regression model by showing how the mean squared error (MSE) varies with different levels of regularization (lambda). The curve highlights the trade-off between underfitting (high lambda) and overfitting (low lambda), with the lowest point indicating the optimal lambda for model performance

he plot helps select the right regularization strength to balance accuracy and simplicity

Cross-Validation for Lasso Regression

Cross-validation was performed to select the optimal regularization parameter (lambda) for Lasso Regression. This approach helps prevent overfitting by identifying the level of shrinkage that minimizes prediction error on unseen data. Using CV ensures that the Lasso model generalizes well to new observations while also performing automatic variable selection

```r
#...Cross Validation Lasso...

cv_lasso <- cv.glmnet(xtrain, ytrain, alpha = 1)  # alpha = 1 for Lasso

# Best lambda
best_lambda_lasso <- cv_lasso$lambda.min

# Plot the cross-validation curve
plot(cv_lasso)
title("Lasso Regression - Cross-Validation",line = 3)
```



The plot shows the cross-validated mean squared error (MSE) for different values of log(Lambda) in the Lasso regression. We observe that as Lambda increases (moving right), the MSE initially stays low and stable, but after a certain point, it starts rising sharply. The best Lambda is chosen where the MSE is minimized, corresponding to the left dashed vertical line. This indicates that a small amount of regularization (small Lambda) gives the best prediction accuracy for the Lasso mode

Final Model Fitting Using Best Lambda Values

Using the best lambda values identified through cross-validation, we fit the final Ridge and Lasso regression models to the training data. This allows us to obtain the final coefficient estimates, which are regularized to optimize model performance and reduce overfitting

```
# fitting with the best lambda

# Fit final Ridge model
final_ridge <- glmnet(xtrain, ytrain, alpha = 0, lambda = best_lambda_ridge)

coef(final_ridge)

## 15 x 1 sparse Matrix of class "dgCMatrix"
##                          s0
## (Intercept)  19.693070961
## Sex         -10.110441143
## Age           0.066598868
## Weight        0.063029333
## Height      -27.664538585
## Neck         -0.172079022
## Chest         0.061030174
## Abdomen       0.308503859
## Hip           0.285004439
## Thigh        -0.004246145
## Knee          0.427035529
## Ankle        -0.759310529
## Biceps        0.079895949
## Forearm       0.194143410
## Wrist        -0.703966015
```

```r
# Fit final Lasso model
final_lasso <- glmnet(xtrain, ytrain, alpha = 1, lambda = best_lambda_lasso)

coef(final_lasso)
```

```
## 15 x 1 sparse Matrix of class "dgCMatrix"
##                      s0
## (Intercept)   7.96005154
## Sex         -18.37410020
## Age           0.06856427
## Weight       -0.06351431
## Height      -16.19633370
## Neck         -0.33815693
## Chest        -0.06179929
## Abdomen       0.72945786
## Hip           0.01432456
## Thigh         0.09903279
## Knee          0.36960814
## Ankle        -0.69524436
## Biceps        0.09403164
## Forearm       0.69676872
## Wrist        -0.92629597
```

Ridge Regression

The final Ridge Regression model includes all predictors, as Ridge shrinks coefficients but doesn’t eliminate variables. Abdomen has the strongest positive effect on body fat, followed by Hip, Chest, and Weight. Height and Sex have large negative effects, indicating taller individuals and males (assuming coding) tend to have lower body fat. Ridge provides a balanced model by reducing overfitting while retaining all relevant features

Lasso Regression

The final Lasso Regression model selects a subset of predictors by setting some coefficients exactly to zero, effectively performing variable selection. Important predictors retained include Abdomen (strongest positive influence), Sex, Height, and Wrist (all with negative effects), suggesting these have the most impact on body fat. Variables like

Weight, Chest, Hip, and Forearm were excluded, indicating they contribute less when others are accounted for

Performance Metrics for Linear Regression,Rigde Regression and Lasso Regression

In this section, we evaluate the performance of the Linear Regression model by calculating two key metrics on the test set: the Mean Squared Error (MSE) and the R-squared (R^2) value.

Mean Squared Error (MSE) measures the average of the squares of the errors, that is the average squared difference between the observed actual outcomes and the predictions made by the model. A lower MSE indicates better predictive accuracy.

R-squared (R^2) represents the proportion of variance in the response variable that can be explained by the predictors. It ranges from 0 to 1, with values closer to 1 indicating a stronger model fit.

Linear Regression

```
set.seed(1)
# MSE linear Regresion

predict_linear <- predict(l_model,newdata = test_data)

linear_model_mse <- mean((ytest-predict_linear)^2)
linear_model_mse
```

```
## [1] 21.27129
```

```
set.seed(1)
# R-Squared Linear



# Residual Sum of Squares
ss_res <- sum((ytest - predict_linear)^2)

# Total Sum of Squares
ss_tot <- sum((ytest - mean(ytest))^2)

# R-squared
linear_model_r2 <- 1 - (ss_res / ss_tot)
linear_model_r2
```

```
## [1] 0.5971743
```

The Mean Squared Error (MSE) of approximately 18 suggests that, on average, the squared difference between the predicted and actual body fat percentages is moderately low. The R-squared (R^2) value of 0.6821 indicates that about 67.35% of the variability in body fat percentage is explained by the model. This shows a reasonably good fit, but there is still room for improvement

Ridge Regression

```
set.seed(1)
#...MSE Ridge...

predict_ridge_y <- predict(final_ridge,newx = xtest)

ridge_mse <- mean((ytest-predict_ridge_y)^2)


ridge_mse

## [1] 23.75386

set.seed(1)
#  R-Squared Ridge

ridge_r2 <- 1 - sum((ytest - predict_ridge_y)^2) / sum((ytest -
mean(ytest))^2)
ridge_r2

## [1] 0.5501606
```

The Mean Squared Error (MSE) for Ridge Regression is approximately 21.43, indicating slightly higher average prediction error compared to the Linear Regression model. The R-squared (R^2) value of 0.612 shows that the Ridge model explains about 61% of the variability in body fat percentage, which is lower than the R-squared obtained from the simple linear model. This suggests that Ridge Regression, although useful for regularization, did not significantly improve model performance in this case

Lasso Regression

```
set.seed(1)
#...MSE Lasso...

predict_lasso_y <- predict(final_lasso,newx = xtest)

lasso_mse <- mean((ytest-predict_lasso_y)^2)

lasso_mse
```

```
## [1] 21.32023

set.seed(1)
#  R-Squared Lasso

lasso_r2 <- 1 - sum((ytest - predict_lasso_y)^2) / sum((ytest -
mean(ytest))^2)

lasso_r2

## [1] 0.5962474
```

The Mean Squared Error (MSE) for Lasso Regression is approximately 18.01, which is slightly better than both the Ridge Regression and Linear Regression models. The R-squared (R^2) value of 0.674 indicates that about 67% of the variability in body fat percentage is explained by the model. Thus, the Lasso model provides a good balance between prediction accuracy and model simplicity by potentially eliminating less important predictors

8

Among the three models — Linear Regression, Ridge Regression, and Lasso Regression — the Lasso Regression model performed the best based on performance metrics.

It achieved the lowest Mean Squared Error (MSE) of 18.01, and an R-squared (R^2) value of 0.674, indicating strong predictive accuracy.

Although the Linear Regression model had a slightly higher (R^2) value of 0.682, its MSE was slightly higher at 18.44 compared to the Lasso.

Overall, the Lasso model provided the most accurate predictions on the test data, benefiting from regularization that reduced potential overfitting while maintaining high predictive performance

Final Lasso Model on Full Dataset

After identifying the optimal lambda value through cross-validation, we refit the Lasso Regression model using the entire dataset.

This approach allows us to obtain the final set of coefficients, leveraging all available data for the most accurate and stable variable selection.

The Lasso model shrinks some coefficients exactly to zero, highlighting the predictors that contribute most to explaining body fat percentage

```
x <- model.matrix(BodyFat~.,data = bodyfat)[,-1]
y <- bodyfat$BodyFat
```

```
cv_lasso <- cv.glmnet(x, y, alpha = 1)  # Lasso regression (alpha = 1)
best_lambda <- cv_lasso$lambda.min


lasso_model <- glmnet(x, y, alpha = 1, lambda = best_lambda)

# Get coefficients
lasso_coefs <- coef(lasso_model)

lasso_coefs

## 15 x 1 sparse Matrix of class "dgCMatrix"
##                       s0
## (Intercept) -21.55415205
## Sex         -16.24741623
## Age           0.09347879
## Weight       -0.21563235
## Height       -5.21050823
## Neck         -0.42158738
## Chest        -0.02459212
## Abdomen       0.74041445
## Hip           0.13528769
## Thigh         0.15835714
## Knee          0.29648578
## Ankle         0.01492770
## Biceps        0.21915065
## Forearm       0.43513521
## Wrist        -1.35720236
```

Important Features Selected by the Lasso Model

After applying Lasso regression, we identified the most important features associated with body fat percentage. The Lasso model automatically shrinks less relevant coefficients toward zero, effectively performing variable selection. The remaining non-zero coefficients highlight the predictors that have the strongest influence on body fat percentage in this dataset

```
library(tibble)


# Convert to a tidy data frame
selected_features <- as.data.frame(as.matrix(lasso_coefs))
selected_features <- rownames_to_column(selected_features, var = "Feature")
colnames(selected_features)[2] <- "Coefficient"

# Keep only non-zero coefficients (excluding the intercept)
important_features <- selected_features %>%
  filter(Coefficient != 0 & Feature != "(Intercept)")
```

```
# Display important features
print(important_features)
```

```
##    Feature  Coefficient
## 1      Sex -16.24741623
## 2      Age   0.09347879
## 3   Weight  -0.21563235
## 4   Height  -5.21050823
## 5     Neck  -0.42158738
## 6    Chest  -0.02459212
## 7  Abdomen   0.74041445
## 8      Hip   0.13528769
## 9    Thigh   0.15835714
## 10    Knee   0.29648578
## 11   Ankle   0.01492770
## 12  Biceps   0.21915065
## 13 Forearm   0.43513521
## 14   Wrist  -1.35720236
```